

```
/*10.Hafta Konuları
CASE YAPISI - Güncelleme*/
/*Örnek:
Personellerin maaşı 5000 den az ise %20,
maaşlar 5000-7000 arasında ise %15,
maaşlar 7000 den fazla ise %10 zam yaparak güncelleyin.*/
```

```
use sirket
update personel
set maas=(
case
    when maas<5000 then maas*0.2+maas --maas*1.2
    when maas>5000 and maas<7000 then maas*0.15+maas
    when maas>7000 and maas<8000 then maas*0.1+maas
    else maas*0.05+maas
end )
from personel
select ad,soyad,maas from personel
```

```
--WHILE DONGUSU
--5 ER artırma örneği
declare @toplaml int
declare @sayac int
set @toplaml=0
set @sayac=1
while @sayac<=10
begin
    set @sayac+=1
    set @toplaml+=5 --@toplaml=@toplaml+5
    print (cast (@toplaml as varchar(3)))
end
```

```
/*ornek:personelin ortalama maaşına bakarak maaş
ortalaması 5000 den az ise
personellere %20 zam yapan bir program yazın.*/
select avg(maas) from personel
select 'ortalama maaş:' + cast (avg(maas) as varchar(10))
from personel
while (select avg(maas) from personel)<6000
begin
```

```
update personel
set maas=maas*1.2
select ad, soyad, maas from personel
end
select 'son ortalama maas:' + cast (avg(maas) as
varchar(10)) from personel
```

/*Break: While döngüsü sonlanmadan herhangi bir işlem basamağında döngüden çıkmak için kullanılır. Özellikle istenilen bir koşul sağlandığında döngüyü sonlandırmak için tercih edilir. Döngü sonlandıktan sonra while döngüsünün bittiği noktanın sonrasında bulunan kodlar çalışmaya devam eder.

Continue: While döngüsü devam ederken bulunan işlem basamağının atlanıp bir sonraki basamaktan döngüye devam edilmesini sağlar.*/

/*ornek: Personel tablosundaki maaşları, maas ortalamasını 2000 seviyesine düşene kadar maaşları %20 azaltan t-sql ifadesini yazın. azaltma işlemi yapılırken en yüksek personel maaşı 2500 ün altına düşmesin.*/

```
select 'ortalama maas:' + cast (AVG(maas) as varchar(10))
FROM PERSONEL
while (select avg(maas) from personel) > 2000
begin
    update personel
    set maas=maas-maas*0.2 --maas*0.8
    select ad, soyad, maas from personel
    if (select max(maas) from personel) <= 2500
    break
end
select 'son ortalama maas:' + cast (AVG(maas) as
varchar(10)) FROM PERSONEL
```

```
/*1 den 15 e kadar olan sayıları yazdırın. 11 i atlayarak yazılsın.*/
```

```
declare @sayac int
set @sayac=0
while @sayac<15
begin
select @sayac+=1
if (@sayac=11)
continue --(buraya break yazarak programı tekrar çalıştırın)
print cast (@sayac as varchar(2))
end
```

```
--GOTO etiketi
```

```
/*kullanıcı tarafından tanımlanan etiketler için koşulsuz dallanmayı sağlar.*/
```

```
/*ornek:1 den 4 e kadar sayılar*/
```

```
declare @sayac int
set @sayac=1
yenile:
print 'sayac degeri: '+cast (@sayac as varchar(2))
set @sayac+=1
while @sayac<=4
goto yenile
```

```
--RETURN İFADESİ
```

```
--bu ifade hazırlanan t-sql kodlarının çalışmasını sonlandırır.
```

```
/*ornek:iki değişkenin içeriğine göre kodların çalışmasını kontrol edilsin.*/
```

```
declare @deger1 int
declare @deger2 int
set @deger2=123
if @deger1 is null
begin
print 'deger1 değişkeni Null'
return --return ifadesini ekleyip kaldırarak programın sonuçlarını inceleyin.
end
else
begin
print @deger1
```

```
    end
If @deger2 is not null
print @deger2
```

--Odev

```
/*1.while döngüsü kullanarak 1-10 arasındaki sayıların
karesini yazdıran bir program yazınız.*/
/*2.while döngüsü ile 3 den başlayarak 3 er artan programı
102 ye kadar yazın.
/*3.döngü ile 1-50 arasındaki sayıların küpünü alan bir
program yazın.
```